

# The Logic of Reachability

David E. Smith

NASA Ames Research Center  
Mail stop 269-2  
Moffett Field, CA 94035  
de2smith@email.arc.nasa.gov

Ari K. Jónsson

Research Institute for Advanced Computer Science  
NASA Ames Research Center  
Mail stop 269-2  
Moffett Field, CA 94035  
jonsson@email.arc.nasa.gov

## Abstract

In recent years, Graphplan style reachability analysis and mutual exclusion reasoning have been used in many high performance planning systems. While numerous refinements and extensions have been developed, the basic plan graph structure and reasoning mechanisms used in these systems are tied to the very simple STRIPS model of action.

In 1999, Smith and Weld generalized the Graphplan methods for reachability and mutex reasoning to allow actions to have differing durations. However, the representation of actions still has some severe limitations that prevent the use of these techniques for many real-world planning systems.

In this paper, we 1) separate the logic of reachability from the particular representation and inference methods used in Graphplan, and 2) extend the notions of reachability and mutual exclusion to more general notions of time and action. As it turns out, the general rules for mutual exclusion reasoning take on a remarkably clean and simple form. However, practical instantiations of them turn out to be messy, and require that we make representation and reasoning choices.

## Introduction

In 1995, Blum and Furst introduced a method for reachability analysis in planning [2, 3]. The method involves incremental construction of a plan graph to provide information about which propositions and actions are possible at each time step. Since then, plan graph analysis has been a key part of several high performance planning systems such as IPP [18], STAN [19], and Blackbox [16]. More recently, reachability analysis has been used for another purpose – to help compute more accurate heuristic distance estimates for guiding state-space search [4, 11, 24, 22] and guiding search in partial-order planners [23].

Reachability analysis and mutual exclusion reasoning have also been the subject of both efficiency improvements [19, 6], and extensions to deal with things like limited forms of uncertainty [26, 28], and resources [17]. Unfortunately, the basic plan graph structure and reasoning mechanisms are limited to the very simple STRIPS model of action. In STRIPS, one cannot talk about time – actions are considered to be instantaneous, or at least of unit duration, preconditions must hold at the beginning actions, and effects are true in the subsequent state. Many real world planning problems require a much richer notion of time and action; actions can

have differing durations, preconditions may need to hold over some or all of the actions, effects may take place at differing times, and exogenous events or conditions may occur.

In 1999, Smith and Weld [27] generalized the Graphplan methods for reachability and mutex reasoning to allow actions to have differing durations. However, the representation of actions used by Smith and Weld still made a number of simplifying assumptions:

1. All effects take place at the end of an action.
2. Preconditions that are unaffected by an action hold throughout the duration of the action.
3. Preconditions that are affected by an action are undefined throughout the duration of the action.
4. There are no exogenous events.

Unfortunately, these restrictions are not reasonable for many real-world domains [14, 25]. Many actions have resource consumption effects that occur at the beginning of the action. Others have effects that are transient. In addition, some action preconditions need only hold at the beginning of an action, or for a limited period. As an example that illustrates all of these, turning a spacecraft involves firing thrusters for periods at the beginning and end of the turn. As a result, there are transient needs for various resources (valves, controllers), transient effects like vibration and heat that occur near the beginning and end, and outright resource consumption (fuel) that occurs near the beginning, and near the end.

Finally, exogenous events are crucial in many domains. For example, in planning astronomical observations, celestial objects are only above the horizon during certain time windows, and they must not be occluded by other bright objects.

While Smith and Weld's Temporal Graphplan (TGP) planner performs extremely well<sup>1</sup>, the representation cannot be easily extended to remove the above restrictions. In particular, when exogenous events and/or transient effects are permitted, reachability and mutual exclusion relationships hold over intervals of time. For example, the action of observing a particular celestial object is only reachable during the intervals when the object is visible. A second problem

---

1. Do [8] and Haslum [10] have reported that TGP continues to outperform more recent domain-independent temporal planners.

with TGP is that the mutex rules are complex, and it has been difficult to verify that they are sound.

In this paper we extend the notions of reachability and mutual exclusion reasoning to deal with the deficiencies in TGP, namely: 1) actions with general conditions and effects, and 2) exogenous conditions. Note that our objective here is not to develop a planning system that does this reasoning, but rather to lay down a formal set of rules for doing this reasoning. Given such a set of rules, there are choices concerning how much reachability reasoning one actually wants to do, which in turn leads to different possibilities data structures, implementations, and search strategies.

In the next section we introduce notation for time and actions. Using this notation, we then develop the laws for simple reachability without mutual exclusion. We then develop a very general but simple set of laws for mutual exclusion reasoning. Finally, we discuss practical issues of implementing these laws. In particular, we discuss some possible restrictions that one might want to impose on mutex reasoning and discuss how these laws can be implemented using a constraint network and generalized arc-consistency techniques.

## The Basics

### Propositions, Time and Intervals

To model many real world planning domains, we need to talk about propositions (fluents) holding at particular points in time, and over intervals of time. We will use the notation  $p:t$  to indicate that fluent  $p$  holds at time  $t$ . We will use the notation  $p:i$  to indicate that  $p$  holds over the interval  $i$ . Thus:

$$p:i \Leftrightarrow \forall (t \in i) p:t$$

We use the standard notation  $[t_1, t_2]$ ,  $(t_1, t_2)$ ,  $(t_1, t_2]$ ,  $[t_1, t_2)$  to refer to closed, open, and partially open intervals respectively, and use  $i^+$  and  $i^-$  to refer to the left and right endpoints of an interval. For our purposes, we do not need a full set of interval relations, such as those defined by Allen [1]. However, we do need the simple relation *meet*. Two intervals *meet* if the right endpoint of the first is equal to the left endpoint of the second, and the common endpoint is contained in at least one of the two intervals (they can't be both open):<sup>2</sup>

$$\text{Meets}(i, j) \Leftrightarrow i^+ = j^- \wedge i^+ \in i \cup j$$

Finally, we use  $i \parallel j$  to refer to the concatenation of two intervals that meet.

### Actions

In many real world domains, actions take time. In order for an action to be successful, certain conditions may need to hold over part or all of the action. Furthermore, different effects of the action may not all occur at the same time. In fact some of these effects may be *transient* – that is, they are only temporarily true during the action. For example, an action

may use a resource (such as a piece of equipment) but release it at the end. In this case the resource becomes unavailable during the action, but becomes available again at the end of the action. To capture all of this, we model actions as having a condition and an effect, both of which are a conjunction of literals.<sup>3</sup> Thus, an action is represented as:

$$\begin{array}{ll} a:t & \text{cond: } p_1:i_1 \wedge \dots \wedge p_n:i_n \\ & \text{eff: } e_1:j_1 \wedge \dots \wedge e_n:j_n \end{array}$$

Where we require that:

1. the conjunction of the condition and effect is logically consistent
2. each effect must start at or after time  $t$ , that is:  $i_k^-, j_k^- \geq t$
3. each of the intervals  $i_k, j_k$  is relative to the start time  $t$ , that is  $i_k, j_k = t + \Delta$ , where the interval  $\Delta$  is not a function of  $t$ .

A simple STRIPS action with preconditions  $p_1, \dots, p_n$  and effects  $e_1, \dots, e_n$  would be modelled as:

$$\begin{array}{ll} a:t & \text{cond: } p_1:t \wedge \dots \wedge p_n:t \\ & \text{eff: } e_1:t+1 \wedge \dots \wedge e_n:t+1 \end{array}$$

As a more complex example, consider an action that requires that  $p$  hold throughout the action, and requires a resource  $r$  for two time units before releasing  $r$  and producing its final effect  $e$ . This would be modelled as:

$$\begin{array}{ll} a:t & \text{cond: } p:[t, t+2] \wedge r:t \\ & \text{eff: } \neg r:(t, t+2) \wedge r:t+2 \wedge e:t+2 \end{array}$$

So what exactly are the semantics of these more general actions? In STRIPS, an action can only be performed if its preconditions hold. In that case, the effects will hold at the next time point or *state*. However, this does not make sense for our more general notion of action, because the condition might specify that a proposition hold at some time after the start of the action. In other words, there is nothing to prevent us from initiating such an action even though part of the condition is not valid. As a result, the semantics we ascribe to actions is that if action  $a:t$  is performed at time  $t$  and all of the conditions hold over the designated time intervals, then the effects will hold over the designated time intervals. If the conditions do not hold, then the outcome of the action is unknown.

Note that there is a subtle difference between the effects:  $e:[t, t') \wedge \neg e:t'$ ,  $e:[t, t')$ , and  $e:t \wedge \neg e:t'$ . The first specifies that  $e$  holds over the designated interval, and ceases to hold after that. The second says that  $e$  holds over the specified interval but may persist after that if nothing else interferes. The

2. We permit the endpoint to be in both intervals. Technically this would be considered overlap by Allen [1].

3. Disjunctions in the condition can be handled by breaking the action into simpler actions with only conjunctive conditions. We could also allow any number of condition/effect pairs, as is done with conditional effects in the PDDL language. However, for our purposes it is more convenient to have different names for each condition/effect pair. As a result, we will suppose that an action with multiple condition/action pairs is broken up into separate actions having disjoint conditions.

last specifies  $e$  at  $t$  and  $t'$  but leaves the status of  $e$  at intermediate times subject to persistence or interference by other actions. All three of these turn out to be useful, but the first is generally the most common.

For convenience we will use  $\text{Cond}(a;t)$  and  $\text{Eff}(a;t)$  to refer to the condition and effect for action  $a;t$  respectively. It is not particularly important how we define the duration of an action, but in keeping with the usual intuitions, we will define it as being the difference between the end of the last effect, and the start of the action. Thus:

$$D(a;t) \equiv \max_{\{j : \text{Eff}(a;t) \Rightarrow e;j\}} j^+ - t$$

## Exogenous Conditions

In order to model more realistic planning problems, we need to model *exogenous conditions*. By an exogenous condition, we mean any condition dictated by actions or events not under the planner's control. For a STRIPS planning problem, the initial conditions are the only type of exogenous conditions permitted. More generally, exogenous conditions can include such things as the intervals during which certain celestial objects are visible, or the times at which resources become available. We can consider exogenous conditions as being the effects of unconditional exogenous actions. For convenience, we will lump all exogenous conditions together, and consider them as being the effects of a single unconditional action,  $X$ :

$$\begin{array}{ll} X;0 & \text{cond:} \\ & \text{eff: } xc_1;i_1 \wedge \dots \wedge xc_n;i_n \end{array}$$

where for initial conditions, the interval would be the time point 0. Thus, for a telescope observation problem, we might have something like:

$$\begin{array}{ll} X;0 & \text{cond:} \\ & \text{eff: Telescope-parked;0} \\ & \quad \wedge \text{Sunset;0023} \\ & \quad \wedge \text{Visible(C842);[0217, 0330]} \\ & \quad \wedge \dots \end{array}$$

For purposes of this paper, we have chosen to consider only unconditional exogenous events. More generally, we might want to consider *conditional* exogenous events – i.e., events that occur only if the specified conditions are met. As it turns out, this extension requires a few additional axioms, but is otherwise not particularly difficult. We will elaborate on this later.

## Simple Reachability

We first consider a very simple notion of reachability; we regard a proposition as being *reachable* at time  $t$  if there is some action that can achieve it at time  $t$ , and each of the conditions for the action is reachable at/over the specified time or interval. This is a very optimistic notion of reachability because even though two conditions for an action might be possible, they might be mutually exclusive, and we are not yet considering this interaction. To formalize reachability,

we will use two modal operators,  $\Diamond(p;t)$ , and  $\Delta(p;t)$ .  $\Diamond(p;t)$ , means that  $p;t$  is logically *possible* – that is,  $p;t$  is consistent with the exogenous conditions.  $\Delta(p;t)$  means that  $p;t$  is optimistically achievable or *reachable* – that is, there is some plan that could (optimistically) achieve  $p;t$ . According to these definitions, if  $p;t$  is reachable, it is possible. However the converse is not true –  $p;t$  can be logically possible, but not reachable, because the set of actions is not sufficiently rich to achieve  $p;t$ .

For convenience, we will allow  $\Diamond$  and  $\Delta$  to apply to intervals as well as single time points:

$$\Diamond(p;i) \equiv \forall(t \in i) \Diamond(p;t)$$

$$\Delta(p;i) \equiv \forall(t \in i) \Delta(p;t)$$

In general, modal logics tend to have nasty computational properties, but the logic we will develop here is particularly simple – we do not require any nesting of these modal operators, and we will not be allowing any quantification inside of a modal operator.

## Exogenous Conditions

The first set of axioms we need are the exogenous conditions. Thus:

$$(\text{Eff}(X;0) \vdash p;t) \vdash p;t \quad (1)$$

Of course, the exogenous conditions are also both possible and reachable:

$$p;i \Rightarrow \Diamond(p;i) \quad (2)$$

$$p;i \Rightarrow \Delta(p;i) \quad (3)$$

Likewise, the negation of any exogenous condition cannot be either possible or reachable:

$$p;i \Rightarrow \neg \Diamond(\neg p;i) \quad (4)$$

$$p;i \Rightarrow \neg \Delta(\neg p;i) \quad (5)$$

Finally, we need to be able to apply the closed world assumption to the exogenous conditions, inferring that anything that is not explicitly prohibited by the initial conditions is possible:

$$(\text{Eff}(X;0) \not\vdash \neg p;t) \vdash \Diamond(p;t) \quad (6)$$

## Persistence

Next, we need a frame axiom for reachability – that is, an axiom that allows us to infer that if a proposition is reachable at a given time then it is reachable later on, just by allowing it to persist. However, we need to make sure that the proposition isn't forced to become false by an exogenous condition. To do this, we require that the proposition also be possible. A first version of this axiom is:

$$\Delta(p;i) \wedge \text{meets}(i, j) \wedge \Diamond(p;j) \Rightarrow \Delta(p;i \parallel j) \quad (7)$$

Here, the intervals  $i$  and  $j$  can be either open or closed – all we require is that they meet. Most commonly,  $i$  will be a single time point  $t$ , and  $j$  an open interval  $(t, t')$ , where  $t'$  is either  $\infty$ , or the next time point at which the proposition  $p$  becomes false because of exogenous conditions.

Unfortunately, this axiom is a bit too optimistic – it allows us to persist transient effects of an action indefinitely

into the future. Normally this is ok, but if an exogenous condition blocks a condition for that action at some time in the future, then the transient effect should not persist indefinitely. For example, suppose that we have a single action  $a;t$  having condition  $p;t$ , and requiring a resource  $r$  for two time units before releasing  $r$  and producing its final effect  $e$ . This would be modelled as:

$$\begin{array}{ll} a;t & \text{cond: } p;t \wedge r;t \\ & \text{eff: } \neg r;(t, t+2) \wedge r;t+2 \wedge e;t+2 \end{array}$$

Now suppose that the conditions  $p$  and  $r$  are initially true, but  $p$  becomes false at time 3. As a result,  $a$  is only reachable up until time 3. The effect  $e$  is first reachable at time 2, but can persist indefinitely. However,  $\neg r$  can only occur during the action, and should therefore only be reachable in the interval  $(0,5)$ . However, Axiom (7) would allow us to persist the reachability of  $\neg r$  indefinitely into the future.

The way we fix this problem is to specialize axiom (7) to only allow action effects to persist if they are not later overridden by the action. Formally, we define  $p;i$  to be a *persistent effect* for an action if there is no other effect  $\neg p;j$  such that  $j$  ends after  $i$ .<sup>4</sup>

$$\text{PersistEff}(a;t) = \left\{ p;i \in \text{Eff}(a;t) : \left( \neg \exists j : j^+ > i^+ \wedge \left( p;j \in \text{Eff}(a;t) \vee \neg p;j \in \text{Eff}(a;t) \right) \right) \right\}$$

Using this definition, we can restrict axiom (7) by requiring that  $p;t$  be a persistent effect:

$$\begin{array}{l} \exists a, t : p;i \in \text{PersistEff}(a;t) \\ \wedge \Delta(p;i) \wedge \text{meets}(i, j) \wedge \Diamond(p;j) \Rightarrow \Delta(p;i \parallel j) \end{array} \quad (8)$$

This allows us to persist the reachability of persistent effects, but not transient ones.

## Actions

Finally, we need axioms that govern when actions are reachable, and what their effects will be. An action is reachable if its conditions are reachable and the effects are not prevented:

$$\Delta\text{Cond}(a;i) \wedge \Diamond(\text{Eff}(a;i)) \Rightarrow \Delta(a;i) \quad (9)$$

Conversely, if an action is reachable, both its conditions and its effects must be reachable:

$$\begin{array}{l} a;t \Rightarrow \text{Cond}(a;t) \wedge \text{Eff}(a;t) \\ \Delta(a;i) \wedge (a;t \Rightarrow p;t') \Rightarrow \Delta(p;t') \end{array} \quad (10)$$

## Conjunctive Optimism

Although Axiom (9) is technically correct, it is difficult to satisfy. The trouble is the premise  $\Delta\text{Cond}(a;t)$ . Typically, the condition for an action will be a conjunction of propositions, so we need to be able to prove that this conjunction is reachable in order to be able to use the axiom. Unfortunately, we cannot usually do this, because our axioms only allow us to

infer that individual effects are possible, (or at best, conjunctions of effects resulting from the same action). Deciding whether a conjunction of propositions is reachable is a planning problem, so there is little hope that we can do it efficiently. Instead, we will be extremely optimistic, and suppose that if the individual propositions are reachable, then the conjunction is reachable:

$$\begin{array}{l} \Delta(p_1;i_1) \wedge \dots \wedge \Delta(p_n;i_n) \\ \Rightarrow \Delta(p_1;i_1 \wedge \dots \wedge p_n;i_n) \end{array} \quad (11)$$

In the next section we will revise this axiom to require that the propositions are not *mutually exclusive*.

## An Example

To see how the axioms for simple reachability work, we return to our example with a single action  $a;t$  having condition  $p;t$ , and requiring a resource  $r$  for two time units before releasing  $r$  and producing its final effect  $e$ :

$$\begin{array}{ll} a;t & \text{cond: } p;t \wedge r;t \\ & \text{eff: } \neg r;(t, t+2) \wedge r;t+2 \wedge e;t+2 \end{array}$$

We suppose that the conditions  $p$  and  $r$  are initially true, but  $p$  becomes false at time 3. We therefore have the exogenous conditions:

$$\begin{array}{ll} X;0 & \text{cond:} \\ & \text{eff: } p;0 \wedge r;0 \wedge \neg p;3 \end{array}$$

Using the axioms developed above, we can now derive reachability for the propositions  $p$ ,  $r$ ,  $\neg r$ ,  $e$ , and the action  $a$ :

- |    |                                               |                    |
|----|-----------------------------------------------|--------------------|
| 1. | $p;0, r;0, \neg p;3$                          | $X;0, (1)$         |
| 2. | $\Delta(p;0), \Delta(r;0)$                    | 1, (3)             |
| 3. | $\Diamond(p;(0, 3)), \Diamond(r;(0, \infty))$ | 1, (6-CWA)         |
| 4. | $\Delta(p;[0, 3]), \Delta(r;[0, \infty))$     | 2, 3, (8-Persist.) |
| 5. | $\Delta(a;[0, 3])$                            | 4, (9)             |
| 6. | $\Delta e;[2, 5), \Delta(\neg r;(0, 5))$      | 5, (10)            |
| 7. | $\Diamond(e;[5, \infty))$                     | 1, (6-CWA)         |
| 8. | $\Delta(e;[2, \infty))$                       | 6, 7, (8-Persist.) |

In this proof the numbers at right refer to the previous lines of the proof, and the axioms that justify the step. A graphical depiction of the final reachability intervals is shown in Figure 1.

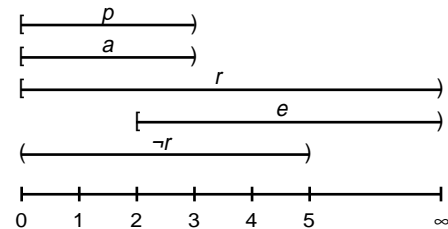


Figure 1: Reachability intervals for a simple example.

Thus, we can see that because the action  $a$  is only possible up until time 3,  $\neg r$  only persists up until time 5, but  $e$  can

4. Since the effects of an action must be consistent, the intervals  $i$  and  $j$  will actually be disjoint.

persist indefinitely. Of course, if there were an exogenous effect that forced  $e$  to be false at some time in the future, then the persistence of  $e$  would also be curtailed by axiom (3). If  $p$  later became true again, we would be able to apply action  $a$  again, so the action  $a$ , and propositions  $e$  and  $\neg r$  could become reachable during additional intervals

The style of reasoning that we have done here closely mimics what goes on in Graphplan – we started at time 0, and worked forward in time, adding new actions and propositions as they became reachable. However, we are not limited to a strict temporal progression – we can draw conclusions in any order, as long as they are sanctioned by the axioms.

## Mutual Exclusion

Much of the power of Graphplan comes from the use of binary mutual exclusion reasoning, which rules out many combinations of incompatible actions and propositions. From the point of view of our logic, proving that two or more actions or propositions are mutually exclusive amounts to proving that the conjunction is not possible and therefore not reachable. We will use an  $n$ -ary modal operator

$$M(p_1; t_1, \dots, p_n; t_n)$$

to indicate that the propositions  $p_1; t_1, \dots, p_n; t_n$  are mutually exclusive. We note that the arguments to  $M$  are commutative and associative. As before we will extend the notation to work on intervals:

$$M(p_1; i_1, \dots, p_n; i_n) \\ \Rightarrow \forall (t_1 \in i_1, \dots, t_n \in i_n) M(p_1; t_1, \dots, p_n; t_n)$$

Using mutual exclusion, we revise the conjunctive optimism axiom (11) to be:

$$\left( \Delta(p_1; i_1) \wedge \dots \wedge \Delta(p_n; i_n) \right) \wedge \neg M(p_1; i_1, \dots, p_n; i_n) \\ \Rightarrow \Delta(p_1; i_1 \wedge \dots \wedge p_n; i_n) \quad (12)$$

Our job then, is to write a set of axioms that allows us to infer when propositions are mutually exclusive. This will restrict what we can infer with axiom (12), and hence restrict our ability to infer when actions are reachable using axiom (9). As in Graphplan, our mutual exclusion laws will be incomplete – we are looking for a set of laws that are computationally effective so that the reasoning can be done in polynomial time. As a result, we will restrict our attention to binary mutual exclusion, noting that if any set of propositions is mutually exclusive, then any superset is mutually exclusive:

$$M(s) \wedge s \subset s' \Rightarrow M(s')$$

As in the work on Temporal Graphplan [27], the fact that we are dealing with a much more general notion of time means that actions and propositions can overlap in arbitrary ways. As a result, it helps to define mutual exclusion between actions and propositions, as well as between pairs of actions and pairs of propositions. In addition, because of exogenous events, and transient action effects, mutual exclu-

sion relationships can come and go repeatedly.<sup>5</sup> As it turns out, the general rules for mutual exclusion reasoning take on a remarkably clean and simple form. However, practical instantiations of them turn out to be more complex.

## Logical mutex

If two propositions are logically inconsistent then it is clearly impossible for them to be true at the same time. Formally:

$$(\psi_1 \Rightarrow \neg \psi_2) \Rightarrow M(\psi_1, \psi_2) \quad (13)$$

where  $\psi_1$  and  $\psi_2$  can be either propositions  $p; t$ , or actions  $a; t$ . This rule is the seed that allows us to infer a number of simple logical mutex relationships. For example, if  $\psi_1 \equiv p; t$  and  $\psi_2 \equiv \neg p; t$  we get the obvious mutex rule:

$$M(p; t, \neg p; t)$$

which forms the basis for Graphplan mutual exclusion reasoning. Similarly, if  $\psi_1 \equiv p; t$ , and  $\psi_2 \equiv a; t'$ , and  $a; t'$  has a precondition or effect  $\neg p; t$ , then the action and proposition are mutex (since  $(a; t' \Rightarrow \neg p; t)$ ):

$$(a; t' \Rightarrow \neg p; t) \Rightarrow M(p; t, a; t')$$

Going a step further, if we have two actions with logically inconsistent preconditions or effects this rule allows us to conclude that the actions are mutex:

$$(a_1; t_1 \Rightarrow p; t) \wedge (a_2; t_2 \Rightarrow \neg p; t) \Rightarrow M(a_1; t_1, a_2; t_2)$$

Although we will not illustrate it here, rule (13) also admits the possibility of inferring additional logical mutex from domain axioms that might be available (e.g. an object cannot be in two places at once). It can also be used to derive logical mutex between actions that have more general resource conflicts.

All of these logical mutex relationships are the seeds that serve to drive the remainder of the mutex reasoning. As we will see below, they allow us to infer additional mutex relationships between actions and propositions, pairs of actions, and ultimately pairs of propositions.

## Implication Mutex

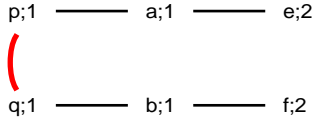
Our second mutex rule is also remarkably simple, but more subtle. If two propositions  $\psi_1$  and  $\psi_2$  are mutex, and some other proposition  $\psi_3$  implies  $\psi_1$ , then  $\psi_3$  is mutex with  $\psi_1$ . Formally:

$$M(\psi_1, \psi_2) \wedge (\psi_3 \Rightarrow \psi_1) \Rightarrow M(\psi_3, \psi_2) \quad (14)$$

Again, the  $\psi_i$  can be either propositions or actions. Suppose that  $\psi_1$  and  $\psi_2$  are mutex propositions, and  $\psi_3$  is an action that has  $\psi_1$  as a precondition. Since the action implies its preconditions, this rule allows us to infer that the action is mutex with  $\psi_2$ . Going one step further, if  $\psi_2$  is an action, then this rule allows us to conclude that the actions  $\psi_3$  and  $\psi_2$  are mutex. Thus, this single rule allows us to move from proposition/proposition mutex to proposition/action mutex, to action/action mutex.

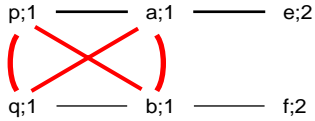
5. In Graphplan and even TGP, once a mutex relationship disappears, it cannot reappear at a later time.

To see how this works, consider two simple STRIPS actions:  $a$ , having precondition  $p$  and effect  $e$ , and  $b$ , having precondition  $q$  and effect  $f$ . Suppose that both  $p$  and  $q$  are reachable at time 1, but that they are mutex as depicted graphically in Figure 2. We can therefore apply the above rule to conclude



**Figure 2:** A simple STRIPS example with  $p$  and  $q$  mutex at time 1.

that  $a;1$  is mutex with  $q;1$  and  $b;1$  is mutex with  $p;1$ . Having done this, we can apply the rule again to conclude that  $a;1$  is mutex with  $b;1$  as shown in Figure 3.



**Figure 3:** Mutex derived by the implication rule

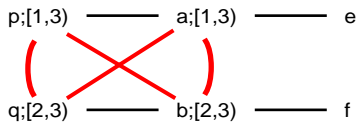
While axiom (14) works fine for a discrete STRIPS model of time, more generally, we do not want to do the mutex reasoning for each individual time point. Instead, we would like to do it for large intervals of time. So suppose we start out with two propositions/actions  $\phi_1$  and  $\phi_2$  being mutex over the intervals  $i_1$  and  $i_2$ , and  $\phi_3; t_3 \Rightarrow \phi_1; t_1$ . Then to find the time interval over which  $\phi_3$  will be mutex with  $\phi_2; i_2$ , we need to gather up all the times  $t_3$  that imply  $\phi_1$  at some point in  $i_1$ . Formally:

$$\begin{aligned} M(\phi_1; i_1, \phi_2; i_2) \wedge i_3 = \{ t : \phi_3; t \Rightarrow \exists (t_1 \in i_1) : \phi_1; t_1 \} \\ \Rightarrow M(\phi_3; i_3, \phi_2; i_2) \end{aligned} \quad (15)$$

To illustrate how this works, we extend our example to continuous time, and imagine that  $p$  and  $q$  are produced by mutually exclusive actions of different duration. In particular, suppose that  $p$  over  $[1,3)$  is mutually exclusive with  $q$  over  $[2,3)$ . Using (15) we could conclude that:

$$\begin{aligned} M(a; [1, 3), q; [2, 3)) \\ M(b; [2, 3), p; [1, 3)) \\ M(a; [1, 3), b; [2, 3)) \end{aligned}$$

as illustrated in Figure 4.



**Figure 4:** Implication mutex for intervals

### Explanatory Mutex

Our final rule is somewhat subtle and tricky – it is, in effect, the explanatory version of the previous rule. Basically, it

says that if all ways of proving  $\psi_1$  are mutex with  $\psi_2$  then  $\psi_1$  and  $\psi_2$  are mutex:

$$\left( \forall \psi_3 \left( (\psi_3 \Rightarrow \psi_1) \Rightarrow M(\psi_3, \psi_2) \right) \right) \Rightarrow M(\psi_1, \psi_2) \quad (16)$$

The tricky part is the phrase “all ways of proving”. For our purposes, we are interested in the case where  $\psi_1$  is a proposition  $p;t$  and  $\psi_3$  is a way of achieving  $p;t$ . We could achieve  $p;t$  by performing an action  $a;t'$  that has  $p;t$  as an effect, but we could also potentially perform the action  $a$  at some earlier time and allow  $p$  to persist. Thus, we need to account for all of these possibilities. Furthermore, if  $p$  is achieved earlier and allowed to persist, that “means of achieving” could be mutex with  $\psi_2$  for one of two reasons: either  $a;t'$  is mutex with  $\psi_2$ , or the persistence of  $p$  is mutex with  $\psi_2$ .

To formalize this, we define the *support* of a proposition as being the union of the *direct support* and the *indirect support* for the proposition:

$$\text{Supp}(p;t) = \text{DirSupp}(p;t) \cup \text{IndSupp}(p;t)$$

The direct support is simply the set of actions that can directly achieve the proposition:

$$\text{DirSupp}(p;t) = \left\{ a;t' : \Delta(a;t') \wedge (\text{Eff}(a;t') \Rightarrow p;t) \right\}$$

The indirect support is a set of miniature plans for achieving the proposition, each consisting of an action  $a;t'$  that achieves the proposition before  $t$ , and the persistence of the proposition until  $t$ . As with persistence axiom (8), we need to be careful not to rely on the persistence of transient effects:

$$\begin{aligned} \text{IndSupp}(p;t) = \left\{ a;t' \wedge p;(t', t) : \right. \\ \left. \Delta(a;t') \wedge t' < t \wedge (\text{PersistEff}(a;t') \Rightarrow p;t'') \wedge \Diamond(p;(t', t)) \right\} \end{aligned}$$

Using this concept of support, we can restate our more specific version of (16) as:

$$\left( \forall \sigma \in \text{Supp}(p;t) : M(\sigma, \psi) \right) \Rightarrow M(p;t, \psi) \quad (17)$$

For the case of direct support,  $\sigma$  is just an action  $a;t$ , so we can directly evaluate  $M(\sigma, \psi)$ . However, for indirect effects,  $\sigma$  is a conjunction of an action  $a;t$  and a persistence  $p;i$ . If either of these is mutex with  $\psi$ , then the conjunction is mutex with  $\psi$ . More generally:

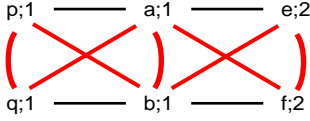
$$M(\sigma_1, \psi) \vee M(\sigma_2, \psi) \Rightarrow M(\sigma_1 \wedge \sigma_2, \psi)$$

As a result, we expand axiom (17) into the more useful form:

$$\begin{aligned} \left( \forall \sigma \in \text{DirSupp}(p;t) : M(\sigma, \psi) \right) \\ \wedge \left( \forall (\alpha \wedge \pi) \in \text{IndSupp}(p;t) : M(\alpha, \psi) \vee M(\pi, \psi) \right) \\ \Rightarrow M(p;t, \psi) \end{aligned} \quad (18)$$

To illustrate how this axiom works, we return to the simple example in Figure 3. From implication mutex we already

know that  $a;1$  and  $b;1$  are mutex. Effect  $e;2$  has only the direct support  $a;1$ . As a result, we can use the above rule to conclude that  $b;1$  is mutex with  $e;2$ . Similarly, we can conclude that  $a;1$  is mutex with  $f;2$ . Finally, using these facts we can conclude that  $e;2$  is mutex with  $f;2$  as shown in Figure 5.



**Figure 5:** Mutex derived by the implication rule

As with Implication Mutex, we would like to be able to apply (17) and (18) to intervals rather than just single time points. If we generalize the notion of support to intervals, we can state the more general version as:

$$(\forall \sigma \in \text{Supp}(p; i_1) : M(\sigma, \varphi; i_2)) \Rightarrow M(p; i_1, \varphi; i_2) \quad (19)$$

As we did with (17) we could expand out to the longer but more useful form containing direct and indirect support.

## Practical Matters

### Limiting mutex reasoning

Although the above mutex theory is very general, it can produce huge numbers of mutex conclusions, many of which would not be very useful. In order to make the reasoning practical, we need to constrain the application of these axioms so that only the most useful mutex relationships are derived.

The first, and most obvious way of limiting the mutex rules is to only apply them to propositions and actions that are actually reachable. If something isn't reachable at a given time, it is mutex with everything else, so there is no point in trying to derive additional mutex relationships.

While this certainly helps, it is not enough. The trouble is that our laws allow us to conclude mutual exclusion relationships for propositions and actions at wildly different times. For example, we might be able to conclude that  $p;2$  is mutually exclusive with  $q;238$ . While this fact could conceivably be useful, it is extremely unlikely. To understand why, and what to do about it, we need to consider how mutex are used.

Fundamentally, we use mutex to decide whether or not the conditions for actions are reachable, and hence whether the actions themselves are reachable (axioms (12) and (9)). Thus, the mutex relations that ultimately matter are the proposition/proposition mutex between conditions for an action. With simple STRIPS actions, this means we are concerned with propositions being mutex at exactly the same time. Unfortunately, with more general conditions we can't do this –

6. In practice, if  $\psi$  is mutex with  $p;t$ , then we do not need to check actions that support  $p$  prior to  $t$  (since the persistence of  $p$  will be mutex with  $\psi$ ). Thus we only need to consider support for  $p$  at times  $t$  after  $p$  is mutex with  $\psi$ . This involves moving the check for persistence mutex back into the definition of independent support.

an action may require  $p;t$ , and  $q;t+1$ . Thus, we'd need to know whether  $M(p;t, q;t+1)$  in order to decide whether the action was reachable. However, we do not care about  $M(p;t, q;t+5)$ . Suppose we define the *separation* for a pair of conditions in an action as the distance between the intervals over which the conditions are required to hold. For our example above, the condition separation was 1. We then take the maximum over all conditions for an action, and the maximum over all actions. This tells us the maximum range of times that we ultimately care about for proposition/proposition mutex relationships. In the extreme case where all pre-conditions of actions are required at the start of the action, we only need to consider whether propositions are mutex at the same time.

We can draw similar conclusions concerning action/action and action/proposition mutex, although in the latter case, the ranges are somewhat wider. This is because we are considering actions that support propositions, which means the actions start before their effects. Still, limiting the application of the axioms to such time ranges drastically reduces the number of mutex conclusions, but with the potential price of missing a few useful mutex relationships. For temporal planning, this tradeoff needs to be carefully investigated.

### Constraint-based reachability reasoning

We now turn our attention to the issue of finding an effective way to calculate reachability information. For this, we turn to constraint reasoning, which is an effective foundation for reasoning about temporal planning problems. The constraint-based reachability reasoning tracks variables that describe reachability, and enforces constraints that eliminate times where actions or propositions are not reachable.

The approach is motivated by the interval representation used for temporal reasoning in various planning systems. In simple temporal network propagation [7], event time domains are described as intervals, and the algorithm is used to infer distance relations between events in plans.

The basic idea appears similar to temporal networks; for each action and proposition, we have a variable representing when it is reachable, and constraints that relate action and proposition reachability. However, this reachability problem does not map to a classical temporal constraint satisfaction problem. This is because action reachability requires necessary conditions to extend over periods of time, so there is no notion of a satisfying assignment to those variables. We therefore turn to a more general class of constraint reasoning problems, where the variables are linked by elimination procedures [12], that specify when intervals can be eliminated from the domains. The result is a network where reachability can be determined effectively by constraint propagation, but there is no notion of a solution to the network. Different constraint propagation methods, such as generalized arc consistency, can be applied to propagate the procedural constraints. A very simple propagation method is to apply the set of elimination procedures to quiescence.

Let  $T$  be the set of possible times, which may be continuous and infinite. Typically,  $T$  will be a sub-interval of the integers or the real numbers. For each action, we define a variable  $a$ , and for each proposition, we define a variable  $p$ . The initial domain of each variable is  $T$ , and the intended semantics are that the variables represent the times at which an action or proposition is reachable.

The simplest reachability procedure enforces that if a fluent is not possible, it is not reachable. This gives rise to the following intervals being eliminated for each variable  $p$ :

$$\{i : \neg\phi(p; i)\}$$

The action reachability axioms are relatively straightforward as well. Let  $a$  be an action, and let  $p_1:d_1, \dots, p_k:d_k$ , be the action conditions, where each  $d_j$  represents the interval distance from the action time. Let  $e_1\delta_1, \dots, e_k\delta_k$  be the action effects, represented with the corresponding relative interval distances.

If a precondition is not reachable at some point within the necessary interval, then the action is not reachable. For each variable  $p_j$ , with eliminated intervals:

$$[a_1, b_1], \dots, [a_k, b_k]$$

we can eliminate from  $a$ :

$$\{[a_j - D, b_j - d] : j \in \{1, \dots, k\}\}$$

where  $d_j = [d, D]$ .

If an effect is not possible, then the achieving action is not reachable. It turns out that we can enforce this in the same way as conditions, as the impossible intervals have already been eliminated from reachability and no other intervals are eliminated from reachability unless no actions can achieve those. For each effect  $e_j$  and each interval  $[a_j, b_j]$  eliminated from  $e_j$ , we can eliminate the interval

$$[a_j - D, b_j - d]$$

from  $a$ , where  $\delta_j = [d, D]$ .

Enforcing the persistence axiom is again more involved. The basic rule states that an interval where  $p$  is not reachable can be extended up to the point where an action can achieve  $p$  or an exogenous event establishes  $p$ . To determine this point, for a given interval, we define the set of subintervals over which an action  $a$  can provide an effect  $p$ :

$$E(a, x, y) = \bigcup \left\{ \begin{array}{l} [s + d, \infty] \cap [x, y], e \in \text{PersistEff}(a;s) \\ [s + d, t + D] \cap [x, y], e \notin \text{PersistEff}(a;s) \end{array} \right\}$$

where the union is over all effects  $e$  of  $a$  and each interval  $[s, t]$  defining the domain of  $a$  for times  $\geq x - D$ . Note that the result is a finite set of intervals.

Let us assume an interval  $[x, y]$  has been eliminated from  $p$ . Let  $z$  be the earliest time after  $y$ , where  $p$  is necessarily true,  $z = \infty$  if there is no such time. Then we can eliminate

the interval that extends from  $y$  to the earliest time where an action can achieve  $p$ . In other words, we can eliminate

$$[y, \min(E(a_1, y, z) \cup \dots \cup E(a_k, y, z))]$$

from the domain of  $a$ , where  $a_1, \dots, a_k$  are all actions that can achieve  $p$ .

Again, we need to extend this notion to allow the elimination of intervals that are not necessarily met by the given unreachable interval, but are nonetheless unreachable, as the reachable conditions in between do not persist. This is easy to do in the interval reasoning framework; we can simply eliminate each interval that is not in the union

$$E(a_1, y, z) \cup \dots \cup E(a_k, y, z)$$

as the persistent effects have already been taken into account. The elimination of the interval immediately following  $[x, y]$  is a special case of this elimination rule.

To see how the application of elimination rules works, we again look at the earlier example. It is given that the following intervals have been eliminated:

$$\begin{array}{l} \neg p; [0] \\ p; [3] \\ \neg r; [0] \\ e; [0] \end{array}$$

Initially, the action condition reachability rules only allow us to eliminate  $a; [3]$ .

Applying the persistence rule to  $p; [3]$ , we calculate  $E(a, 3, \infty)$  and find that it is empty. This allows us to eliminate  $p; [3, \infty]$ . Applying the persistence rule to other eliminated intervals allows us to eliminate:

$$\begin{array}{l} \neg p; [0, 3) \\ e; (0, 2) \end{array}$$

Now that more intervals have been eliminated for  $p$ , the application of the action condition reachability rules allows  $a; [3, \infty]$  to be eliminated.

Finally, calculating  $E(a, 0, \infty)$  for  $\neg r$ , we get  $(0, 5)$ , which allows us to eliminate  $\neg r; [5, \infty]$ . Note that the result is the same as applying the logical axioms to determine when actions and propositions may be reachable.

The above formulation does not include mutual exclusion reasoning. For mutex reasoning, the variables will correspond to pairs of propositions/actions, and the domains will be sets of two dimensional intervals. Using these only requires extending the elimination procedure for action reachability to also eliminate actions where two preconditions are mutually exclusive. Although we understand the basic outline of the elimination procedures for mutex reasoning, we have not yet worked through all the details.



## Discussion

### Exogenous events:

For purposes of this paper we assumed that exogenous events or actions were not conditional in nature. As a result, we lumped all of the exogenous effects together into a single action with no conditions. It is not too difficult to extend our theory to allow general exogenous events. Initially, we start with the set  $X$  of all effects from unconditional exogenous actions. In order for an exogenous event to take place, its conditions must be satisfied. Thus, any exogenous event whose conditions are satisfied in  $X$  will also take place, so its effects must be added to  $X$ . We continue in this way until we obtain the closure of all exogenous conditions. The remaining exogenous actions may or may not occur. However, if their conditions ever become true, they will definitely occur. As a result, we need to treat them like domain axioms. In other words, if  $a;t$  is a conditional exogenous event, we need to add the axioms:

$$\text{Cond}(a;t) \Leftrightarrow a;t \Rightarrow \text{Eff}(a;t)$$

The problem therefore reduces to one of handling domain axioms, which the theory already handles.

## Conclusions

In this paper, we extended reachability and mutual exclusion reasoning to apply to a much richer notion of action and time. In doing this, we provided a formalization of these notions that is independent of any particular planning framework. Surprisingly, the rules for mutual exclusion reasoning turn out to be simpler and more elegant than we expected, particularly given the complexity of the rules for Temporal Graphplan developed by Smith and Weld [27].

There are still a number of issues involved in making this reasoning practical for temporal planning systems. Restricting the intervals over which the mutex rules apply seems critical, but there are tradeoffs in the veracity of the resulting mutex reasoning. Efficient interval representation and reasoning is also crucial. Superficially, the problem of determining reachability looks like it could be cast as a constraint satisfaction problem. However, as we've discussed above, the constraints are complex elimination procedures, and it is not yet clear whether this approach will be computationally effective.

We are continuing to work towards a CSP implementation within the Europa planning system [13, 14] and hope to apply these techniques to real problems involving spacecraft and rovers.

## Acknowledgments

Thanks to Jeremy Frank and Dan Weld for discussions on extending reachability analysis to a richer notion of time. This research was supported by NASA Ames Research Center and the NASA Intelligent Systems program.

## References

- 1 Allen, J. 1984. Towards a general theory of action and time. *Artificial Intelligence* 23(2), 123–154.
- 2 Blum, A., and Furst, M. 1995. Fast planning through planning graph analysis. In *Proc. 14th Int. Joint Conf. on AI*, 1636–1642.
- 3 Blum, A., and Furst, M. 1997. Fast planning through planning graph analysis. *Artificial Intelligence* 90, 281–300.
- 4 Bonet, B. and Geffner, H. 2001. Planning as heuristic search. *Artificial Intelligence* 129, 1–29.
- 5 Brafman, R.I. 2001. On reachability, relevance, and resolution in the planning as satisfiability approach, *J. AI Research* 14, 1–28.
- 6 Cayrol, M., Régnier, P., and Vidal, V. 2000. New results about LCGP, a least committed Graphplan. *Proc. 5th Int. Conf. on AI Planning & Scheduling*.
- 7 Dechter, R., Meiri, I. and Pearl, J. 1991. Temporal constraint networks. *Artificial Intelligence* 49, 61–95.
- 8 Do, M. and Kambhampati, S. 2001. Sapa: a domain-independent heuristic metric temporal planner. *Proc. 6th European Conf. on Planning*.
- 9 Ghallab, M., and Laruelle, H. 1994. Representation and control in IxTeT, a temporal planner. *Proc. 2nd Int. Conf. on AI Planning Systems*, 61–67.
- 10 Haslum, P. and Geffner, H. 2001. Heuristic planning with time and resources. *Proc. 6th European Conference on Planning*.
- 11 Hoffmann, J. and Nebel, B. 2001. The FF planning system: fast plan generation through heuristic search, *J. AI Research* 14, 253–302.
- 12 Jónsson, A. 1997. *Procedural Reasoning in Constraint Satisfaction*. Ph.D. dissertation, Dept. of Computer Science, Stanford U.
- 13 Jónsson, A., Morris, P., Muscettola, N., and Rajan, K. 1999. Next generation remote agent planner. *Proc. 5th Int. Symposium on AI, Robotics and Automation in Space*. ESTEC, Noordwijk, Netherlands, 363–370.
- 14 Jónsson, A., Morris, P., Muscettola, N., Rajan, K., and Smith, B. 2000. Planning in interplanetary space: theory and practice. *Proc. 5th Int. Conf. on AI Planning & Scheduling*.
- 15 Kambhampati, S., Lambrecht, E. and Parker, E., 1997. Understanding and extending Graphplan. *Proc. 4th European Conf. on Planning*.
- 16 Kautz, H. and Selman, B. 1999. Unifying SAT-based and graph-based planning, *Proc. 16th Int. Joint Conf. on AI*.
- 17 Koehler, J. 1998. Planning under resource constraints. *Proc. 13th European Conf. on AI*. 489–493
- 18 Koehler, J., Nebel, B., Hoffmann, J. and Dimopoulos, Y. 1997. Extending planning graphs to an ADL subset. *Proc. 4th European Conf. on Planning*, 275–287.
- 19 Long, D., and Fox, M. 1999. The efficient implementation of the plan graph in STAN. *J. AI Research* 10.
- 20 Mackworth, A. 1977. Consistency in networks of relations. *Artificial Intelligence* 8, 99–118.

- 21 Muscettola, N. 1994. HSTS: integrating planning and scheduling. In Zweben, M., and Fox, M., eds., *Intelligent Scheduling*. Morgan Kaufmann. 169–212.
- 22 Nguyen, X. and Kambhampati, S. 2000. Extracting effective and admissible heuristics from the planning graph. *Proc. 17th National Conf. on AI*.
- 23 Nguyen, X. and Kambhampati, S. 2001. Reviving partial order planning. *Proc. 17th Int. Joint Conf. on AI*.
- 24 Refanidis, I. and Vlahavas, I. 2001. The GRT planning system: backward heuristic construction in forward state-space planning. *J. AI Research* 15, 115–161.
- 25 Smith, D., Frank, J. and Jónsson, A. 1990. Bridging the gap between planning and scheduling. *The Knowledge Engineering Review* 15(1).
- 26 Smith, D. and Weld, D. 1998. Conformant Graphplan. *Proc. 15th National Conf. on AI*, 889–896.
- 27 Smith, D. and Weld, D. 1999. Temporal planning with mutual exclusion reasoning. *Proc. 16th Int. Joint Conf. on AI*, 326–333.
- 28 Weld, D., Anderson, C. and Smith, D. 1998. Extending Graphplan to handle uncertainty & sensing actions. *Proc. 15th National Conf. on AI*, 897–904.